CLAIMS

I claim:

1. A method of decoding comprising an iterative procedure of the steps: (a) quantization (if necessary),

(b) initialization of memory and memory location-address pointers,

(c) application of a recursive table-lookup decoding method for each constituent convolutional subcode while utilizing the appropriate set of pre-stored lookup tables and the appropriate memory locations for each subcode,

(d) application of more iterations of step (c) until a total of L iterations has been completed,

(e) extraction of decoded binary estimates for the information-bits from the memory locations representing the approximations for the maximum a posteriori estimates,

for the decoding of parallel-concatenated convolution codes of fixed blocklength and to produce a block of decoded binary bits that approximate decoded bits from an iterative maximum a posteriori decoder.

2. A method of decoding for parallel-concatenated convolutional codes that consist of two or more binary convolutional constituent subcodes of finite blocklength Z and share a block of Z information-bits with respect to corresponding interleaver-orderings, so as to produce a block of Z decoded binary bits for the information-bits that are an approximation to decoded bits obtained from an iterative maximum a posteriori decoder when initially given code-bit channel-symbol values or quantized digital data representing channel-symbol values, wherein the decoding method is a procedure comprising the steps:

(a) quantization of received code-bit channel-symbol values into digital data (if necessary),

(b) initialization by storing of digital data corresponding to code-bits into appropriately assigned memory locations as well as storing appropriate initialized digital data values at required locations including the memory locations that are shared by all the constituent subcodes to represent the maximum a posteriori reliability estimates which are initialized to digital data representing equi-likely estimates as also are the memory locations for the extrinsic estimates and punctured code-bits,

(c) application of a recursive table-lookup decoding method for each constituent convolutional subcode while utilizing the appropriate set of pre-stored lookup tables and the appropriate memory locations for each subcode,

(d) application of more iterations of step (c) until a total of L iterations has been completed,

(e) extract the decoded binary estimates for the information-bits from the digital data at the shared memory locations representing the maximum a posteriori estimates by utilizing the most significant bit of the digital datas

(where the benefit of U.S. Provisional Patent Application No. 60/391.092 with filing date 06/24/2002 is incorporated).

3. The iterative decoding method of claim 2, wherein the locations/entries of the memory are digital address-words and the memory locations assigned to the types of reliability estimates are digitally-sequentially ordered so that a sequentially incrementing address pointer indicates the Z locations in memory of each type of estimate.

4. The iterative decoding method of claim 2, wherein the recursive table-lookup decoding method is a decoding method for a binary convolutional code of finite blocklength Z so as to produce:

memory updated with digital data representing approximations for the maximum a posteriori reliability estimates for the Z information-bits;

and memory updated with digital data representing approximations for the extrinsic reliability estimates for the Z information-bits,

wherein the decoding method is a recursive table-lookup procedure comprising the steps:

(a) initialization by writing/storing of data into memory,

(b) reading data from memory,

(c) reading data from a set of pre-stored lookup tables,

(d) writing/storing data read from lookup tables into memory,

(e) incrementing a memory location-address pointer,

when initially given memory stored with digital data representing:

the channel-symbol reliability estimates for the Z parity-bits;

previous extrinsic reliability estimates for the Z information-bits;

and maximum a posteriori reliability estimates for the Z information-bits, which is a function of the previous extrinsic estimates, the a priori estimates, and the channel-symbol estimates,

5. The recursive table-lookup decoding method of claim 4, wherein the locations/entries of the data to be read out from lookup tables at some current recursion are determined from:

data read from lookup tables in the current recursion; and/or

data read from lookup tables in previous recursions; and/or

data read from memory which was stored previously when read from a lookup table in a previous recursion; and/or

data read from memory which was initially stored.

6. The recursive table-lookup decoding method of claim 5, wherein the locations/entries of the data to be read out from lookup tables are digital address-words that are formed by appending together one of more digital data-words which are read out from lookup tables and/or memory.

7. The recursive table-lookup decoding method of claim 4, wherein the number of recursions is twice the blocksize, Z, of the convolutional code and the locations/entries of the data to be read

out from memory during a recursion are digital address-words that for the first Z recursions will increment sequentially from an address-word value of zero to an address-word value of (Z minus one) and then for the second Z recursions will increment by decreasing sequentially from (Z minus one) to zero.

8. The recursive table-lookup decoding method of claim 4, wherein the number of recursions is twice the blocksize, Z, of the convolutional code and the locations/entries of the data to be read out from memory during a recursion are digital address-words that for the first Z recursions will increment with respect to a permuted ordering of the digital address-word values of zero to (Z minus one) and then for the second Z recursions will increment through the reverse of the permuted ordering.

9. The recursive table-lookup decoding method of claim 4, wherein the number of seperate lookup tables in the set of lookup tables is a design parameter where seperate lookup tables can be combined to form fewer lookup tables, or seperate lookup tables can be split into several lookup tables.

10. The recursive table-lookup decoding method of claim 4, wherein the digital data-words that are pre-stored into the lookup tables is a design parameter, where the best mode of operation selects pre-stored data values based on selected inherent mathematical/computational functions and selected inherent quantization functions that the lookup-tables are approximating.

11. The recursive table-lookup decoding method of claim 4, wherein the set of lookup tables are pre-stored with digital data-words, based on inherent mathematical/computational functions and quantization functions such that the produced decoded data-words representing an approximation to the maximum a posteriori estimate are approximating a modified version of the maximum a posteriori estimate, including the modification that adds a sensitivity factor to the forward state probabilities and the reverse state probabilities within the inherent functions.

12. A recursive table-lookup decoding method comprising a recursive procedure of the steps:

(a) initialization by writing/storing of data into memory,

(b) reading data from memory,

(c) reading data from a set of pre-stored lookup tables,

(d) writing/storing data read from lookup tables into memory,

(e) incrementing a memory location-address pointer,

for the decoding of binary convolutional codes of fixed blocklength such as to produce a block of estimates which approximate the maximum a posteriori estimates for the information-bits, and to produce a block of estimates which approximate the extrinsic estimates for the information-bits.

13. The recursive table-lookup decoding method of claim 12, wherein a binary convolutional code of finite blocklength Z is being decoded so as to produce:

memory updated with digital data representing approximations for the maximum a posteriori reliability estimates for the Z information-bits;

and memory updated with digital data representing approximations for the extrinsic reliability estimates for the Z information-bits,

when initially given memory stored with digital data representing functions of:

the channel-symbol reliability estimates for the Z parity-bits and the Z information-bits;

and the a priori reliability estimates for the Z information-bits.

14. The recursive table-lookup decoding method of claim 13, wherein the locations/entries of the data to be read out from lookup tables at some current recursion are determined from:

data read from lookup tables in the current recursion; and/or

data read from lookup tables in previous recursions; and/or

data read from memory which was stored previously when read from a lookup table in a previous recursion; and/or

data read from memory which was initially stored.

15. The recursive table-lookup decoding method of claim 14, wherein the locations/entries of the data to be read out from lookup tables are digital address-words that are formed by appending together one of more digital data-words which are read out from lookup tables and/or memory.

16. The recursive table-lookup decoding method of claim 13, wherein the given data representing functions of the reliability estimates are given as: the channel-symbol reliability estimates for the parity-bits;

some given appropriate estimates for the information-bits;

and reliabiltiy estimates that are a combination of the given appropriate estimates for the information-bits, the a priori estimates for the information-bits, and the channel-symbol estimates for the information-bits.

17. The recursive table-lookup decoding method of claim 13, wherein the number of recursions is twice the blocksize, Z, of the convolutional code and the locations/entries of the data to be read out from memory during a recursion are digital address-words that for the first Z recursions will increment sequentially from an address-word value of zero to an address-word value of (Z minus one) and then for the second Z recursions will increment by decreasing sequentially from (Z minus one) to zero.

18. The recursive table-lookup decoding method of claim 13, wherein the number of recursions is twice the blocksize, Z, of the convolutional code and the locations/entries of the data to be read out from memory during a recursion are digital address-words that for the first Z recursions will increment with respect to a permuted ordering of the digital address-word values of zero to (Z

minus one) and then for the second Z recursions will increment through the reverse of the permuted ordering.

19. The recursive table-lookup decoding method of claim 13, wherein the number of seperate lookup tables in the set of lookup tables is a design parameter where seperate lookup tables can be combined to form fewer lookup tables, or seperate lookup tables can be split into several lookup tables.

20. The recursive table-lookup decoding method of claim 13, wherein the digital data-words that are pre-stored into the lookup tables is a design parameter, where the best mode of operation selects pre-stored data values based on selected inherent mathematical/computational functions and selected inherent quantization functions that the lookup-tables are approximating.

21. The recursive table-lookup decoding method of claim 13, wherein the set of lookup tables are pre-stored with digital data-words, based on inherent mathematical/computational functions and quantization functions such that the produced decoded data-words representing an approximation to the maximum a posteriori estimate are approximating a modified version of the maximum a posteriori estimate, including the modification that adds a sensitivity factor to the forward state probabilities and the reverse state probabilities within the inherent functions.